




Test Result	Method being tested	Explanation
 makeValidBooking() 648ms	makebooking(String currentUser, String BookingDate, String BookingTime)	This is a test to see if a booking can be made for a valid user, who has an assigned doctor, and chooses a date and time that is available for both the doctor and the patient. If the method returns true, then the booking was made. As you can see the test result passed so the booking was successfully made.
 checkUnavailability() 2.0s	checkAvailability(String DoctorId, String BookingDate, String BookingTime)	This is a test for the checkAvailability method, When a booking is available the method should return true, but false when unavailable. So using the booking details in the makeValidBooking test, we check to see if a the a booking is available to be made then, we expect the method to return false, meaning unavailable, which it did, so the test passed.
 deleteValidBooking() 390ms	deleteBooking(String patient, String BookingDate, String BookingTime)	This is a test to see if a booking that exists can be deleted. We specifically delete the booking that was made in the previous test, because we know the patient, booking date and time. When a booking is successfully deleted the method returns true, as it did. And you can see the test passed.
 testViewBooking() 91ms	testViewBooking()	This is a test for getTitle(), checks whether the object was created correctly and isn't null and also checks email is expecting <a href="mailto:alice.smith@example.com">alice.smith@example.com</a> . If either of these tests fail the test will then fail.
 testBackWithNull() 860ms	testBackWithNull()	This is a test also for the viewBooking class and checks back() works correctly and throws it a null value, should throw an exception.
 testRegister()	register(String Forename, String Surname, String Email, String	This method enters in valid example data for a brand new account using an email that hasn't been used before. Since this email hasn't been used

	Password)	before the method register should return true, indicating a new user has been registered. The test passed meaning the method worked as intended
 invalidtestRegister()	register(String Forename, String Surname, String Email, String Password)	This is a test to make sure duplicate accounts cannot be created. It is done by using the same data as in the previous test. The register method should return false indicating an account with that email already exists. As you can see the test passed, meaning the method returned false as expected since an account with that email has already been registered
 blankRegister() 372ms	register(String Forename, String Surname, String Email, String Password)	This is a test to make sure that the various fields required to make an account cannot be left empty. The register method should return false if it is unable to create an account. And seeing as the test has passed that mean that the method has worked as intended
 validLoadBookings() 885ms	loadBookings(String inputDate)	This is a test to make sure that the loadbooking method is able to successfully query the booking table in the database for a valid user and get all the bookings for that user. If the loadbooking method is able to successfully query the database then it returns true. Since the method passed this means the method was able to query the database and is working as intended.